Project	Title
gasQS	**gasQS flonic Z1 Modbus Specification**
Status	Keywords
RELEASED	gasQS flonic Z1, Modbus, firmware, specification
Classification	Comments
FOR CUSTOMERS	For non-metrology-controlled gasQS flonic Z1 devices

gasQS flonic Z1 Modbus Specification

Version 2

Documentation for the gasQS flonic V2 firmware version 01.02.10:

Mems AG

Bruggerstrasse

CH-5413 Birmenstorf

Switzerland

Document Identification	File	Date	Number of Pages
MemsDoc20240518TK02	gasQS_flonic_Z1_Modbus_ Specification_Rev02.docx	28.11.2024	57
Department/Office	Author	Phone	Fax
SW	Thomas Kleiner, Florian Krischker	+41 56 470 92 05	+41 56 470 92 09

1 Table of Contents

2 Table of Tables

3 Table of Figures

4 Introduction

This document specifies the MODBUS interface to a gasQS flonic V2 device.

The MODBUS interface specification is described in “Modbus_over_serial_line_V1_02.pdf”, a hardware specific version (OSI levels 1 and 2) of “Modbus_Application_Protocol_V1_1b3.pdf” (OSI level 7).

5 Purpose and Scope

The purpose of this document is to show the MODBUS features which are supported by a gasQS device.

The scope of this document is dual:

- On the one hand, it serves as the specification of the MODBUS interface to be implemented in a gasQS device.

- On the other hand, this is a reference for the interface supported by a gasQS device to determine which MODBUS functions can be controlled by a MODBUS master.

6 MODBUS Specification for gasQS Devices

6.1 Conventions

The following words are used to define the significance of each particular requirement.

- “MUST” / “REQUIRED”
  - All requirements containing the word "MUST" are mandatory. The word MUST, or the adjective "REQUIRED", means that the item is an absolute requirement of the implementation.
- “SHOULD” / “RECOMMENDED”
  - All recommendations containing the word "SHOULD", or the adjective “RECOMMENDED”, are considered desired behaviour. These recommendations should be used as a guideline when choosing between different options to implement functionality. There may be valid reasons in particular circumstances to ignore these recommendations, but the full implications should be understood and the case carefully weighed before choosing a different course.
- “MAY” / “OPTIONAL”
  - The word “MAY”, or the adjective "OPTIONAL", means that this item is truly optional. One designer may choose to include the item because a particular marketplace requires it or because it enhances the product, for example; another designer may omit the same item.

Since a gasQS device satisfies the MUST requirements but not all the SHOULD requirements, it is “conditionally compliant” to the Modbus over serial line specification [1] and the Modbus application protocol specification [2].

6.2 MODBUS Protocol Principle

A MODBUS communication uses a master-slave scheme. There is only one master and up to 247 slaves.

A MODBUS communication is always initiated by the master, i.e. a slave does not send any message without being requested by the master; a slave may not communicate with any other slave. The master initiates only one request at the same time.

A master request uses one of two modes:

- Unicast
  - The master sends a request to one slave;
  - The addressed slave sends an answer to the master but not in case of a frame error. In that case, the slave does not answer at all, whereas in case of any other error, the slave sends an error message back;
  - Each slave must have a unique address (1-247).
- Broadcast
  - The master sends a request to all slaves;
  - None of the slaves sends any response;
  - A broadcast is therefore a writing command;
  - All devices must accept the broadcast writing instruction;
  - Address 0 is reserved to identify a broadcast exchange.

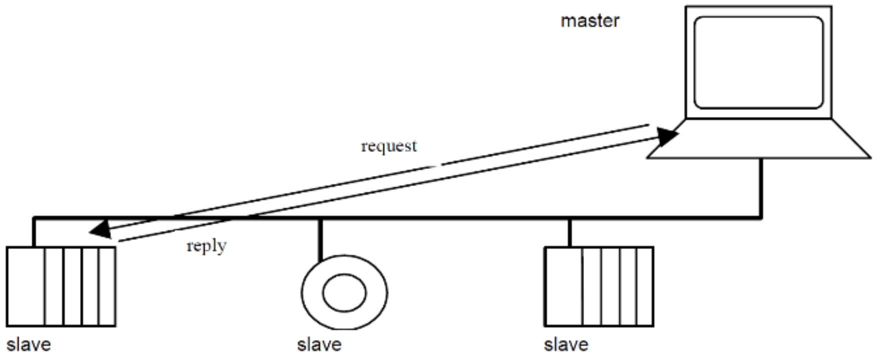

Figure 1: Unicast mode

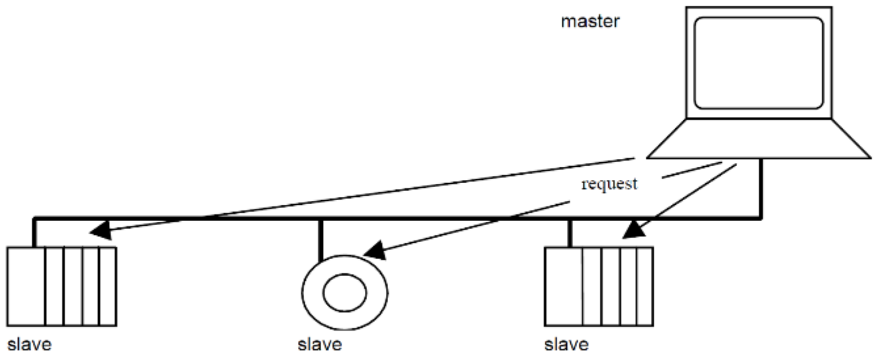

Figure 2: Broadcast mode

6.3 Slave State Diagram

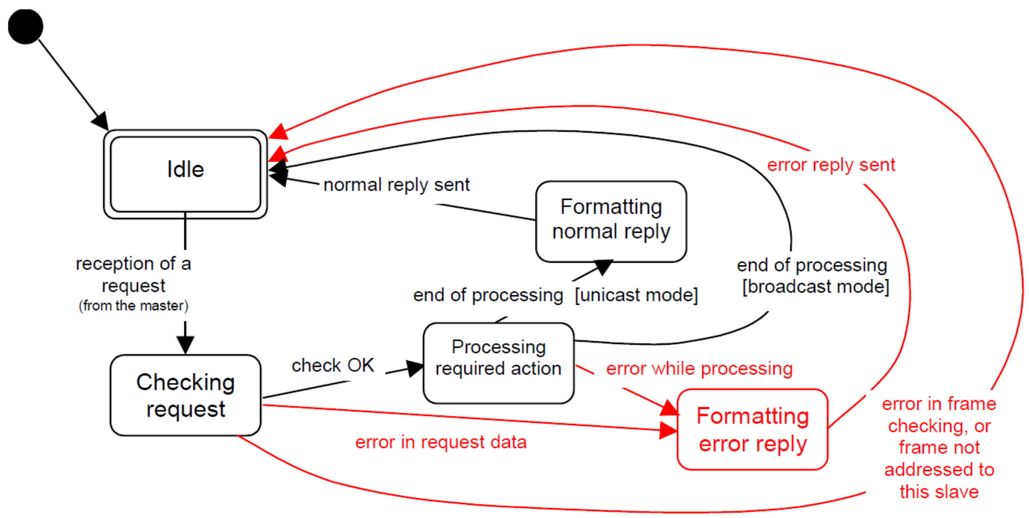

Figure 3: Slave state diagram

In case of an error in frame checking or an invalid slave device address, the device will not send a response. The slave will never respond to broadcast messages, even not in an error case. Error messages will be sent in case of errors in 'request data' (PDU) are detected or if an error occurs during processing the request.
MODBUS diagnostics information is defined and may be requested by the supported diagnostics functions.

6.4 MODBUS Communication Time Diagram

The following figure shows the time diagram of 3 typical scenarios of Master / Slave communications. Section “Exchange i-1” represents the case when a slave sends either a positive answer or an error message back to the master. Section “Exchange i” shows a broadcast message from the master to all slaves who must not respond. Section “Exchange i+1” represents the case of a frame error, the slave does not send any answer back, the master must therefore timeout to end waiting.

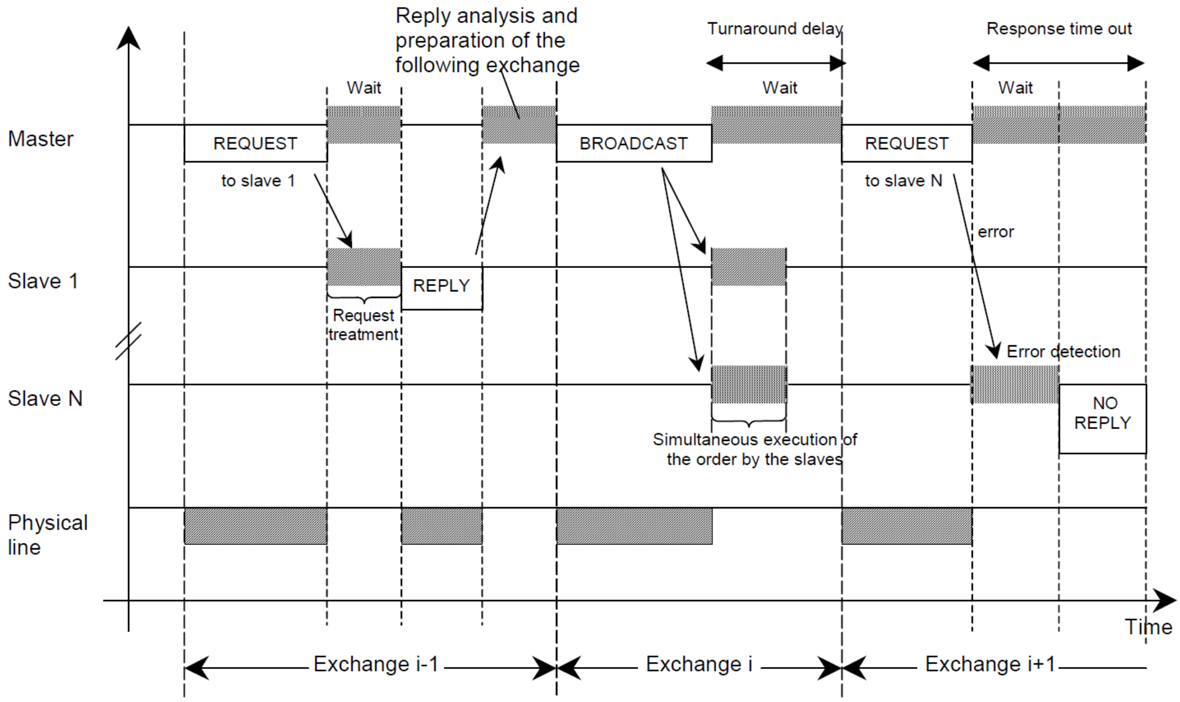

Figure 4: Master / Slave scenario time diagram

6.5 MODBUS Addressing Rules

0	From 1 to 247	From 248 to 255
Broadcast address	Slave individual addresses	Reserved

Table 1: MODBUS Addressing

All slave nodes must recognize the broadcast address.
The master has no specific address.

6.6 MODBUS Frame Description

6.6.1 Introduction

All gasQS devices starting from version 2 implement RTU Mode. The optional ASCII mode is also supported.

6.6.2 RTU Transmission Mode

Each frame byte is coded in hexadecimal, i.e. it contains two 4-bit hexadecimal digits. Each message must be transmitted in a continuous stream of characters. The maximum time gap between two bytes in the same frame is 1.5 characters. If exceeded, the message frame is declared incomplete and should be discarded by the receiver.
The minimum time gap between two frames is 3.5 characters.
For baud rates greater than 19200 bps, fixed time values are used: a value of 750µs for the inter-character time-out (t1.5) and a value of 1.750ms for inter-frame delay (t3.5).

6.6.2.1 Byte Format (11-bit)

Coding System: 8–bit binary
Bits per Byte: 1 start bit
8 data bits, least significant bit sent first
1 bit for parity completion
1 stop bit

6.6.2.2 Serial Transmission of Characters

Even parity is default. In order to ensure maximum compatibility with other products, it is possible to change to **odd** and **no parity** modes.

With Parity Checking										
Start	1 (LSB)	2	3	4	5	6	7	8 (MSB)	Parity	Stop

Table 2: Modbus RTU serial transmission of characters with parity checking

Without Parity Checking										
Start	1 (LSB)	2	3	4	5	6	7	8 (MSB)	Stop	Stop

Table 3: Modbus RTU serial transmission of characters without parity checking

6.6.2.3 RTU Frame Description

<table>
<tr><th colspan="5">Modbus Serial Line ADU[1]</th></tr>
<tr><td>Address Field</td><td>Function Code</td><td>Data Field</td><td colspan="2">CRC</td></tr>
<tr><td rowspan="2">1 byte</td><td rowspan="2">1 byte</td><td rowspan="2">0 up to 252 byte(s)</td><td colspan="2">2 bytes</td></tr>
<tr><td>CRC Low</td><td>CRC High</td></tr>
<tr><td></td><td colspan="2">Modbus PDU[2]</td><td colspan="2"></td></tr>
</table>

Table 4: RTU Frame

The **Address field** contains the **slave address**.
When the master addresses a specific device (unicast), it uses any of the 1-247 slave address. The slave answers with the same address, to let the master know who is answering.

[1] Application Data Unit
[2] Protocol Data Unit

When the master broadcasts, it uses address 0. None of the slave sends any answer to the master.
The **function code** indicates what kind of action the slave(s) must carry out. The function code can be followed by a **data field** that contains additional parameters needed to perform the request.
When answering, the slave sends the same **function code** back to the master if there is no error ("positive MODBUS response"), and possibly additional **data**, depending on the function that was performed.
In case of an error, the slave sends the same function code back, with the MSB set to indicate a fault ("MODBUS exception response"). It adds a one-byte exception code in the data field for a more detailed error description.

Data within a device may be accessed via one of four primary tables, i.e. address spaces, using the appropriate access **function code**. Those tables are:

- Discrete Input, i.e. bit input
- Coils, i.e. bit input/output
- Input Registers, i.e. 16b input
- Holding Registers, i.e. 16b input/output

To address any element(s) of those above-listed tables, the **data field** may be split in two parts: the device table element[3] address (often called the **data address**) and the data itself. The **data address** range is 0-65535[4, 5] for each of the four tables[6].
The **data field** encoding is big-endian, i.e. any 16b value is sent with MSB first.

The RTU mode includes a frame error–checking field that is based on **a Cyclical Redundancy Checking** (CRC) method (see Appendix B MODBUS over Serial Line Specification and Implementation Guide V1.02 [1]), which is performed on the message contents (excluding CRC). For the CRC generation details refer to the MODBUS specification and implementation guide [1], chapter 6.2.1.
After receiving a frame from the master, the slave verifies each byte's parity bit and computes the CRC. If any parity bit or the CRCs are mismatching, the slave does not answer and the master listening times out. The use of the parity bit is configurable. The default settings are parity even and one stop bit. The combination parity odd with one stop bit and no parity with 2 stop bits are supported. The CRC is always used.

6.6.3 ASCII Transmission Mode

The ASCII transmission mode is less efficient for data transmission than the RTU mode, but it has no timing requirements. Every transmitted byte is represented as two 7-bit ASCII characters.

---

[3] Elements are numbered starting from 1, whereas data addresses are starting from 0, so one must add one to the address to get the element number.
[4] Note that this address is the MODBUS address, not that of the device HW. The device provides for the internal mapping between MODBUS data addressing space and device HW resource addressing space.
[5] Due to limited frame length, public functions may only access a limited range within this space, e.g. read coils may only read up to 2000 coils at a time.
[6] The addresses can appear in several tables independently of each other and relate to different device functions.

6.6.3.1 Byte Format (11-bit)

Coding System: 7–bit binary
Bits per Byte: 1 start bit
7 data bits, least significant bit sent first
1 bit for parity completion
1 stop bit

6.6.3.2 Serial Transmission of Characters

Even parity is default. In order to ensure maximum compatibility with other products, it is possible to change to **odd** and **no parity** mode.

With Parity Checking									
Start	1 (LSB)	2	3	4	5	6	7 (MSB)	Parity	Stop

Table 5: Modbus ASCII serial transmission of characters with parity checking

Without Parity Checking									
Start	1 (LSB)	2	3	4	5	6	7 (MSB)	Stop	Stop

Table 6: Modbus ASCII serial transmission of characters without parity checking

6.6.3.3 ASCII Frame Description

Modbus Serial Line ADU[1]					
Start	Address	Function Code	Data	LRC	End
1 char :	2 chars	2 chars	0 up to 2x252 char(s)	2 chars	2 chars CR, LF
		Modbus PDU[2]			

Table 7: ASCII Frame

In ASCII mode, messages include an error–checking field that is based on a **Longitudinal Redundancy Checking** (LRC) method. The LRC field checks the contents of the message, exclusive the beginning 'colon' character and ending CRLF pair. It is applied regardless of any parity checking method used for the individual characters of the message. For the LRC generation details refer to the MODBUS specification and implementation guide [1], chapter 6.2.1.

6.7 Function codes

6.7.1 Function Code Categories

There are three categories of MODBUS Functions codes. They are:

Public Function Codes

- Are well defined function codes,
- guaranteed to be unique,
- validated by the MODBUS.org community,
- publicly documented,
- have available conformance test,
- include both defined public assigned function codes as well as unassigned function codes reserved for future use.

User-Defined Function Codes

- there are two ranges of user-defined function codes, i.e. 65 to 72 (0x41 to 0x48) and from 100 to 110 decimal (0x64 to 0x6E).
- the user can select and implement a function code that is not supported by the specification.
- there is no guarantee that the use of the selected function code will be unique
- if the user wants to re-position the functionality as a public function code, he must initiate an RFC to introduce the change into the public category and to have a new public function code assigned.
- MODBUS Organization, Inc expressly reserves the right to develop the proposed RFC.

Reserved Function Codes

- Function Codes currently used by some companies for legacy products and that are not available for public use.

A MODBUS gasQS device does not use any reserved function codes. It mostly uses a subset of public function codes and a few private function codes for Mems' internal use. See the next paragraph for details.

6.7.2 Function Codes used by MODBUS gasQS Devices

The following table shows the public function codes supported by a MODBUS gasQS flonic device.

6.7.2.1 Public Function Code Definition

Function Code		Primary Table	Object Type
Code	Sub-Code		
0x01		Read Coils	Bit access
0x02		Read Discrete Inputs	Bit access
0x03		Read Holding Registers	16 bits access
0x04		Read Input Registers	16 bits access
0x05		Write Single Coil	Bit access
0x06		Write Single Register	16 bits access
0x08	0x00	Diagnostics	
0x0F		Write Multiple Coils	Bit access
0x10		Write Multiple Registers	16 bits access
0x2B	0x0E	Read device Identification	

Table 8: gasQS Public Function Codes

6.8 Modbus Function Codes & Primary Table Address Maps

The following table shows the primary tables available to access the various gasQS resources.

6.8.1 Coils (Function Codes0x01, 0x05 and 0x0F)

For the function codes 0x01 and 0x0F the state of one coil is represented by a single bit value TRUE = 1 and FALSE = 0. For the function code 0x05 the state of the coil is represented by a 16-bit value TRUE = 0xFF00 and FALSE = 0x0000.

Type	Function codes	Sub code	Address	Access type	Description
Coils	0x01, 0x05, 0x0F	**Basic**			
			0x0000	w	Run measurement sequence (start = TRUE; ignore = FALSE)[7]
			0x0000	r	Measurement sequence running status (busy = TRUE; ready = FALSE)
			0x0001	w	Run purge sequence (start = TRUE; ignore = FALSE)[7]
			0x0001	r	Purge sequence running status (busy = TRUE; ready = FALSE)
			0x0002	w	Run pressurize sequence (start = TRUE; ignore = FALSE)[7]
			0x0002	r	Pressurize sequence running status (busy = TRUE; ready = FALSE)
			0x0003	w	Run depressurize sequence (start = TRUE; ignore = FALSE)[7]
			0x0003	r	Depressurize sequence running status (busy = TRUE; ready = FALSE)
			0x0004	w	Run gas temperature measurement (start = TRUE; ignore = FALSE)
			0x0004	r	Gas temperature measurement running status (busy = TRUE; ready = FALSE)
			…		Reserved
			0x0010	w	Set ISO OUT1 (on = TRUE; off = FALSE)
			0x0010	r	Get ISO OUT1 status (on = TRUE; off = FALSE)
			0x0011	w	Set ISO OUT2 (on = TRUE; off = FALSE)
			0x0011	r	Get ISO OUT2 status (on = TRUE; off = FALSE)
			0x0012	w	Set ISO OUT3 (writing has no effect)
			0x0012	r	Get ISO OUT3 status (on = TRUE; off = FALSE)
			…		Reserved
			0x1000	w	Reset error code register (reset = TRUE; NOP = FALSE)
			0x1000	r	Error code register status (Error = TRUE; no error = FALSE)
			0x1001	w	Reset device microcontroller

[7] Information on allowing the user to abort measurements can be found in the settings under coil address 0x1003.

					(reset = TRUE; NOP = FALSE)
			0x1001	r	Returns always FALSE
			0x1002	w	Run LED test (start = TRUE; stop = FALSE)
			0x1002	r	LED test running status (busy = TRUE; ready = FALSE)
			0x1003	w	Abort any sequence (abort = TRUE; ignore = FALSE)[7]
			0x1003	r	Abort any sequence (busy = TRUE; ready = FALSE)
			0x1004	w	Save all data before power off the device (save = TRUE; NOP = FALSE)
			0x1004	r	(pending = TRUE; ready = FALSE)
			…		Reserved
		Administration, password protected			
			0x8000	wp	Set input valve
			0x8000	r	Get input valve state
			0x8001	wp	Set output valve
			0x8001	r	Get output valve state
			…	…	Reserved
			0x8004	wp	Data factory reset[8] (reset = TRUE; NOP = FALSE)
			0x8004	r	Returns always FALSE
			0x8005	wp	Error counters reset (reset = TRUE; NOP = FALSE)
			0x8005	r	Returns always FALSE
			…	…	Reserved
			0x8200	wp	Run adjustment procedure[9] (start = TRUE; ignore = FALSE)
			0x8200	r	Adjustment procedure running status[9] (busy = TRUE; ready = FALSE)
			0x8201	wp	Set adjustment procedure pending flag[9] (pending = TRUE; NOP = FALSE)
			0x8201	r	Read adjustment procedure pending flag[9] (pending = TRUE; not pending = FALSE)
			…	…	Reserved

Table 9: Coils address mapping

[8] No data relevant for the adjustment procedure is reset for devices with "ADJUSTMENT_MODE_AUTO" (see chapter 7.3).
[9] The adjustment procedure is not available by default. This option needs to be preconfigured at the factory.

6.8.2 Discrete Inputs (Function Code 0x02)

For the function code 0x02 the state of one coil is represented by a single bit value TRUE = 1 and FALSE = 0.

Type	Function codes	Sub code	Address	Access type	Description
Discrete inputs	0x02		0x0000	r	Device ready (ready = TRUE; failed = FALSE)[10]
			0x0001	r	Flow sensor ready (ready = TRUE; failed = FALSE)
			0x0002	r	Pressure sensor ready (ready = TRUE; failed = FALSE)
			0x0003	r	Vdda supply range: 3.00V to 3.45V (OK = TRUE; error = FALSE)
			0x0004	r	+12V supply range: 8.4V to 14.1V (OK = TRUE; error = FALSE)
			0x0005	r	+5V supply range: Obsolete Returns always TRUE
			0x0006	r	Isolated +3.3V: Vcc > 3.0V (OK = TRUE; error = FALSE)
			0x0007	r	Isolated GPIO: Vcc > 2.25V (OK = TRUE; error or not connected = FALSE)
			0x0008	r	Controller temperature in range: -20°C to 60°C (OK = TRUE; failed = FALSE)
			0x0009	r	Sensor temperature in range: -10°C to 55°C (OK = TRUE; failed = FALSE)
			0x000A	r	Input pressure in range: Only updated during measurements: > 3.575bara (OK = TRUE; failed = FALSE)
			0x000B	r	BLD checksum (OK = TRUE; failed = FALSE)
			0x000C	r	APP checksum (OK = TRUE; failed = FALSE)
			0x000D	r	CFG checksum (OK = TRUE; failed = FALSE)
			0x000E	r	Main clock quartz ready (OK = TRUE; error = FALSE)
			…	…	Reserved

Table 10: Discrete inputs address mapping

[10] Combined information: This Discrete Input comes TRUE if all relevant checks were successful. The state of the isolated GPIO (address 0x0007) and the input pressure (address 0x000A) is ignored).

6.8.3 Holding Registers (Function Codes0x03, 0x06, 0x10)

Some settings require writing 32bit values into two linked holding registers. New settings are only accepted if both the factor register and decimal power register, have been written successfully for the power of ten representation, or, if both MSW and LSW registers, have been written successfully for floating point numbers.

Type	Function codes	Sub code	Address	Access type	Data Type	Range min./ max.	Description	
Holding registers	0x03, 0x06, 0x10	**Basic**						
			0x0000	r/w[11]	UINT16	0..65535	Repetition rate in seconds[12] (unsigned values, repetition OFF = 0, repetition ON = 1 to 65535, default: 30 seconds)	
			0x0001	r/w[11]	Reserved	UINT16	Purge cycles (unsigned values, 0x0000 to 0x00FF, default: 0x0000)	
			0x0002	r/w	UINT16	0x01..0xF7	Modbus slave address (0x01 to 0xF7)	
			0x0003	r/w[13]	UINT16	0..6	RS485 baud rate (default: 1)	
							0	9600 bps
							1	19200 bps
							2	38400 bps
							3	57600 bps
							4	115200 bps
							5	230400 bps
							6	460800 bps
			0x0004	r/w[13]	UINT16	0..2	RS485 parity bit (default: 1)	

[11] Needs to finish last cycle before new setting is activated. Set the repetition rate to OFF = 0 before entering a new time value.
[12] By setting a repetition rate, the first measurement is started immediately.
[13] Reset required to activate new setting.

							0	2 stop bits
							1	even parity bit + 1 stop bit
							2	odd parity bit + 1 stop bit
			0x0005	r/w [13]	UINT16	0..1	Mode (default: 0)	
							0	RTU
							1	ASCII
			0x0006	r/w [13]	UINT16	0x00..0xFF	ASCII Input Delimiter (default: '\n')	
			0x0007	r/w	UINT16	0..1000	Additional Modbus response delay in milliseconds (unsigned values, 0 to 1000, default: 0)	
			0x0008	r/w	UINT16	1..8	Correlation values average length (unsigned values, 1 to 8, default: 8)	
			…	…	…	…	Reserved	
			0x0010	r/w	INT16	-32768.. 327670	Abs. pressure P_m[14] in mbara	
			0x0011	r/w	INT16	-32768.. 327670	Abs. pressure P_m[14] in mbara decimal power	
			0x0012	r/w	INT16	-32768.. 327670	Temperature T_m[14] in K	
			0x0013	r/w	INT16	-32768.. 327670	Temperature T_m[14] in K decimal power	
			…	…	…	…	Reserved	
			0x0100	w	UINT16	0x0000.. 0xFFFF	Admin password 0 (Unlock administrator mode)	
			0x0100	r	UINT16	0x0000, 0xFF00	Admin mode (inactive = 0x0000; active = 0xFF00)	
			0x0101	w	UINT16	0x0000.. 0xFFFF	Admin password 1 (Unlock administrator mode)	
			0x0101	r	UINT16	0x0000, 0xFF00	Admin mode (inactive = 0x0000; active = 0xFF00)	

[14] At metering conditions where the gas quantity is measured (flow/volume meter).

			0x0102	w	UINT16	0x0000.. 0xFFFF	Admin password 2 (Unlock administrator mode)
			0x0102	r	UINT16	0x0000, 0xFF00	Admin mode (inactive = 0x0000; active = 0xFF00)
			0x0103	w	UINT16	0x0000.. 0xFFFF	Admin password 3 (Unlock administrator mode)
			0x0103	r	UINT16	0x0000, 0xFF00	Admin mode (inactive = 0x0000; active = 0xFF00)
			…	…	…	…	Reserved
			0x0110	r/w	UINT16	0x0000.. 0xFFFF	Username 0[15]
			0x0111	r/w	UINT16		Username 1
			0x0112	r/w	UINT16		Username 2
			0x0113	r/w	UINT16		Username 3
			0x0114	r/w	UINT16		Username 4
			0x0115	r/w	UINT16		Username 5
			0x0116	r/w	UINT16		Username 6
			0x0117	r/w	UINT16		Username 7
			0x0118	r/w	UINT16		Username 8
			0x0119	r/w	UINT16		Username 9
			0x011A	r/w	UINT16		Username 10
			0x011B	r/w	UINT16		Username 11
			0x011C	r/w	UINT16		Username 12
			0x011D	r/w	UINT16		Username 13
			0x011E	r/w	UINT16		Username 14
			0x011F	r/w	UINT16		Username 15
			0x0120	r/w	UINT16	0..9999	Date: Year (0…9999)[15]
			0x0121	r/w	UINT16	1..12	Date: Month (1…12)
			0x0122	r/w	UINT16	1..31	Date: Day of month (1…31)

[15] Username, date and time values are stored to the audit trail / event log / adjustment log, whenever an adjustment procedure was run.

			0x0123	r/w	UINT16	0..23	Time: Hours (0…23)[15]
			0x0124	r/w	UINT16	0..59	Time: Minutes (0…59)
			0x0125	r/w	UINT16	0..59	Time: Seconds (0…59)
			…	…	…	…	Reserved
			0x0200	r	INT16	read only	Analysis counter (Correlation register 0) adjustment factor coefficient[16]
			0x0201	r	INT16	read only	Analysis counter (Correlation register 0) adjustment factor decimal power exponent[16]
			0x0202	r	INT16	read only	Correlation register 1 adjustment factor coefficient[16]
			0x0203	r	INT16	read only	Correlation register 1 adjustment factor decimal power exponent[16]
			0x0204	r	INT16	read only	Correlation register 2 adjustment factor coefficient[16]
			0x0205	r	INT16	read only	Correlation register 2 adjustment factor decimal power exponent[16]
			0x0206 – 0x021F	r	INT16	read only	Correlation register 3 – 15 adjustment factor coefficient and decimal power exponent[16] as above
			…		…		Reserved
			0x1000 – 0x100F	r/w	UINT16	0x0000.. 0xFFFF	Free user data registers
			…	…	…	…	Reserved
			0x7010	r/w	float32 MSW	$2^{-126}..2^{+127}$	Abs. pressure P_m[14] in mbara (floating-point MSW)
			0x7011	r/w	float32 LSW	$2^{-126}..2^{+127}$	Abs. pressure P_m[14] in mbara (floating-point LSW)
			0x7012	r/w	float32 MSW	$2^{-126}..2^{+127}$	Temperature T_m[14] in K (floating-point MSW)
			0x7013	r/w	float32 LSW	$2^{-126}..2^{+127}$	Temperature T_m[14] in K (floating-point LSW)
			…	…	…	…	Reserved
			0x7200	r	float32 MSW	read only	Analysis counter (Correlation register 0) adjustment factor (floating-point MSW)[16]

[16] Adjustment factor values are read only for gasQS flonic NG devices, and Z1 devices with automatic adjustment enabled. They have no effect on counters or error codes.

			0x7201	r	float32 LSW	read only	Analysis counter (Correlation register 0) adjustment factor (floating-point LSW)[16]		
			0x7202	r	float32 MSW	read only	Correlation register 1 adjustment factor (floating-point MSW)[16]		
			0x7203	r	float32 LSW	read only	Correlation register 1 adjustment factor (floating-point LSW)[16]		
			0x7204	r	float32 MSW	read only	Correlation register 2 adjustment factor (floating-point MSW)[16]		
			0x7205	r	float32 LSW	read only	Correlation register 2 adjustment factor (floating-point LSW)[16]		
			0x7206 – 0x721F	r	float32 MSW	read only	Correlation register 3 – 15 adjustment factor (floating-point MSW and LSW)[16] as above		
			…	…	…	…	Reserved		
		Administration, password protected							
			0x8000	r/wp	UINT16	0..5	Base conditions (default: 0)		
							Register value	Output value (Tn / Tb)	Density (Tn)
							0	0 °C / 25 °C	0 °C
							1	0 °C / 0 °C	0 °C
							2	20 °C / 25 °C	20 °C
							3	15 °C / 15 °C	15 °C
							4	0 °C / 15 °C	0 °C
							5[17]	user specific	user specific
			0x8001	r/wp	UINT16	0..4	Units (default: 0)		
							Register value	Output value	Density
							0	MJ / m^3	kg / m^3
							1	kWh / m^3	kg / m^3
							2	kcal / m^3	kg / m^3

[17] For customer specific produced devices only.

							3	BTU / ft^3	lb / ft^3
							4[17]	user specific	user specific
			0x8002	r/wp[13]	UINT16	2000.. 60000	Max. measurement start delay in milliseconds to fill the pressure chamber (2000 to 60000; default: 3000)		
			0x8003	r/wp	UINT16	0..3	Error auto reset (default: 2)		
							Register value	Error reset mode	
							0	On request only (coil)	
							1	On read-out (input register)	
							2	On retry	
							3	On retry and read-out	
			0x8004	r/wp	UINT16	0..3	Exception reference (default: 2)		
							Register value	Exception reference source	
							0	None	
							1	Measurement counter	
							2	Analysis counter	
							3	Operating hours	
			…	…	…	…	Reserved		
			0x8007	rp/wp[13]	UINT16	1000.. 60000	Time to vent the vacuum chamber (pump start-up relief) in milliseconds (default: 3000)[18]		
			0x8008	rp/wp[13]	UINT16	0.. 10000	Pre-start delay in milliseconds to protect the vacuum pump (default: 100)[18]		
			0x8009	rp/wp[13]	UINT16	500.. 10000	Post-start delay in milliseconds to let the vacuum pump starting-up (default: 500)[18]		
			0x800A	rp/wp[13]	UINT16	1000.. 60000	Max. evacuation delay in milliseconds to evacuate the system (default: 10000)[18]		

[18] Takes effect on low pressure systems with vacuum pump only.

			0x800B	rp/wp[13]	UINT16	1000.. 60000	Time delay in milliseconds to allow the gas exchange inside the measurement system (default: 3000)[18]
			…	…	…	…	Reserved
			0x8100	rp/wp	UINT16	0x0000.. 0xFFFF	Set admin password 0
			0x8101	rp/wp	UINT16	0x0000.. 0xFFFF	Set admin password 1
			0x8102	rp/wp	UINT16	0x0000.. 0xFFFF	Set admin password 2
			0x8103	rp/wp	UINT16	0x0000.. 0xFFFF	Set admin password 3
			…	…	…	…	Reserved
			0x8400	r/w	UINT32 MSW	0x0000.. 0xFFFF	Logfile pointer (unsigned integer address MSW)
			0x8401	r/w	UINT32 LSW	0x0000.. 0xFFFF	Logfile pointer (unsigned integer address LSW)
			…	…	…	…	Reserved

Table 11: Holding registers address mapping.

6.8.4 Input Registers (Function Code 0x04)

There are two ways for reading and interpreting the measurement results:

1. From address 0x0000 to 0x0025 the measurement results from the correlation registers are transmitted in scientific notation as pairs of signed integers for coefficient and exponent to the powers of ten. This method is compliant with the standard Modbus.

$$y = m \cdot 10^n$$

Where:
$m = Correlation\ register\ coefficient$
$n = Correlation\ register\ decimal\ power\ exponent$

2. From address 0x7000 to 0x7025 the measurement results from the correlation registers are presented in the 32bit single precision floating-point format according to the IEEE-754 standard (ANSI/IEEE Std 754-1985; IEC-60559:1989 – International version). Each floating-point value requires two holding registers.

Type	Function codes	Sub code	Address	Access type	Data Type	Description
Input registers	0x04	**Basic**				
			0x0000	r	INT16 (UINT32 MSW)	Analysis counter MSW (default) (Correlation register 0, UINT32 MSW)
			0x0001	r	INT16 (UINT32 LSW)	Analysis counter LSW (default) (Correlation register 0, UINT32 LSW)
			0x0002	r	INT16	Correlation register 1 coefficient[19]
			0x0003	r	INT16	Correlation register 1 decimal power exponent[19]
			0x0004	r	INT16	Correlation register 2 coefficient[19]
			0x0005	r	INT16	Correlation register 2 decimal power exponent[19]
			0x0006 – 0x001F	r	INT16	Correlation register 3 – 15 coefficient and decimal power exponent as above[19]

[19] Correlation assignment for non-standard devices.

			0x0020	r	INT16	Lambda ratio coefficient
			0x0021	r	INT16	Lambda ratio decimal power exponent
			0x0022	r	INT16	Pressure ratio coefficient
			0x0023	r	INT16	Pressure ratio decimal power exponent
			0x0024	r	INT16	Flow ratio coefficient
			0x0025	r	INT16	Flow ratio decimal power exponent
			0x0026	r	INT16	Chamber pressure value in mbara coefficient (updated automatically)
			0x0027	r	INT16	Chamber pressure value in mbara decimal power exponent
			0x0028	r	INT16	Chamber temperature value coefficient (updated during measurement)
			0x0029	r	INT16	Chamber temperature value decimal power exponent
			…	…	…	Reserved
			0x0200	r	INT16	Analysis counter (Correlation register 0) adjustment factor coefficient[20]
			0x0201	r	INT16	Analysis counter (Correlation register 0) adjustment factor decimal power exponent[20]
			0x0202	r	INT16	Correlation register 1 adjustment factor coefficient[20]
			0x0203	r	INT16	Correlation register 1 adjustment factor decimal power exponent[20]
			0x0204	r	INT16	Correlation register 2 adjustment factor coefficient[20]
			0x0205	r	INT16	Correlation register 2 adjustment factor decimal power exponent[20]
			0x0206 – 0x021F	r	INT16	Correlation register 3 – 15 adjustment factor coefficient[20] and decimal power exponent[20] as above
			…		…	Reserved
			0x0220	r	INT16	Lambda ratio adjustment factor coefficient[21]
			0x0221	r	INT16	Lambda ratio adjustment factor decimal power exponent[21]
			0x0222	r	INT16	Pressure ratio adjustment factor coefficient[21]
			0x0223	r	INT16	Pressure ratio adjustment factor decimal power exponent[21]
			0x0224	r	INT16	Flow ratio adjustment factor coefficient[21]

[20] The value of the input register corresponds to the values of the holding register at the same address 0x02xx.
[21] This value is stored to the audit trail / event log / adjustment log, whenever an adjustment procedure was run.

			0x0225	r	INT16	Flow ratio adjustment factor decimal power exponent[21]
			…		…	Reserved
			0x0300	r	INT16	Lower measurement limit register 0 coefficient
			0x0301	r	INT16	Lower measurement limit register 0 decimal power exponent
			0x0302	r	INT16	Lower measurement limit register 1 coefficient
			0x0303	r	INT16	Lower measurement limit register 1 decimal power exponent
			0x0304	r	INT16	Lower measurement limit register 2 coefficient
			0x0305	r	INT16	Lower measurement limit register 2 decimal power exponent
			0x0306 – 0x031F	r	INT16	Lower measurement limit register 3 – 15 coefficient and decimal power exponent as above
			…		…	Reserved
			0x0400	r	INT16	Upper measurement limit register 0 coefficient
			0x0401	r	INT16	Upper measurement limit register 0 factor power exponent
			0x0402	r	INT16	Upper measurement limit register 1 coefficient
			0x0403	r	INT16	Upper measurement limit register 1 factor power exponent
			0x0404	r	INT16	Upper measurement limit register 2 coefficient
			0x0405	r	INT16	Upper measurement limit register 2 factor power exponent
			0x0406 – 0x041F	r	INT16	Upper measurement limit register 3 – 15 coefficient and decimal power exponent
			…		…	Reserved
			0x1000	r	UINT16	Error code register[21]
			0x1001	r	UINT16	Total errors counter
			0x1002	r	UINT32 MSW	Error reference (unsigned long MSW)
			0x1003	r	UINT32 LSW	Error reference (unsigned long LSW)
			0x1004	r	UINT16	Error counter ADC
			0x1005	r	UINT16	Error counter ADC recalibration
			0x1006	r	UINT16	Error counter I2C

			0x1007	r	UINT16	Error counter EEPROM access
			0x1008	r	UINT16	Error counter flow sensor
			0x1009	r	UINT16	Error counter pressure sensor
			0x100A	r	UINT16	Error counter OS unknown
			0x100B	r	UINT16	Error counter OS signal timeout
			0x100C	r	UINT16	Error counter calibration
			0x100D	r	UINT16	Error counter valve
			0x100E	r	UINT16	Error counter upstream pressure
			0x100F	r	UINT16	Error counter downstream pressure
			0x1010	r	UINT16	Error counter device not ready
			0x1011	r	UINT16	Error counter measurement
			0x1012	r	UINT16	Error counter evaluation
			0x1013	r	UINT16	Error counter Modbus
			0x1014	r	UINT16	Error counter pressure operation mode
			0x1015	r	UINT16	Error counter adjustment
			0x1016	r	UINT16	Error counter Flash CRC
			0x1017	r	UINT16	Error counter power supply
			…	…	…	Reserved
			0x1200	r	UINT32 MSW	Adjustment alarm status flag register MSW[21]
			0x1201	r	UINT32 LSW	Adjustment alarm status flag register LSW[21]
			0x1202	r	UINT32 MSW	Adjustment time stamp (value stored as operating hours) MSW[21]
			0x1203	r	UINT32 LSW	Adjustment time stamp (value stored as operating hours) LSW[21]
			0x1204	r	UINT32 MSW	Adjustment interval in days MSW
			0x1205	r	UINT32 LSW	Adjustment interval in days LSW
			…	…	…	Reserved
			0x7000	r	INT16 (UINT32 MSW)	Analysis counter MSW (default)[19] (Correlation register 0, UINT32 MSW)
			0x7001	r	INT16 (UINT32 LSW)	Analysis counter LSW (default)[19] (Correlation register 0, UINT32 LSW)

			0x7002	r	float32 MSW	Correlation register 1, floating-point MSW[19]
			0x7003	r	float32 LSW	Correlation register 1, floating-point LSW[19]
			0x7004	r	float32 MSW	Correlation register 2, floating-point MSW[19]
			0x7005	r	float32 LSW	Correlation register 2, floating-point LSW[19]
			0x7006 – 0x701F	r	float32 MSW & float32 LSW	Correlation register 3 – 15, floating-point MSW and LSW) as above[19]
			0x7020	r	float32 MSW	Lambda ratio (floating-point MSW)
			0x7021	r	float32 LSW	Lambda ratio (floating-point LSW)
			0x7022	r	float32 MSW	Pressure ratio (floating-point MSW)
			0x7023	r	float32 LSW	Pressure ratio (floating-point LSW)
			0x7024	r	float32 MSW	Flow ratio (floating-point MSW)
			0x7025	r	float32 LSW	Flow ratio (floating-point LSW)
			0x7026	r	float32 MSW	Chamber pressure value (updated automatically; floating-point MSW)
			0x7027	r	float32 LSW	Chamber pressure value (floating-point LSW)
			0x7028	r	float32 MSW	Chamber temperature value (updated during measurement; floating-point MSW)
			0x7029	r	float32 LSW	Chamber temperature value (floating-point LSW)
			…	…	…	Reserved
			0x7200	r	float32 MSW	Analysis counter (Correlation register 0) adjustment factor (floating-point MSW)[22]
			0x7201	r	float32 LSW	Analysis counter (Correlation register 0) adjustment factor (floating-point LSW)[22]
			0x7202	r	float32 MSW	Correlation register 1 adjustment factor (floating-point MSW)[22]
			0x7203	r	float32 LSW	Correlation register 1 adjustment factor (floating-point LSW)[22]
			0x7204	r	float32 MSW	Correlation register 2 adjustment factor (floating-point MSW)[22]
			0x7205	r	float32 LSW	Correlation register 2 adjustment factor (floating-point LSW)[22]
			0x7206 – 0x721F	r	float32 MSW & float32 LSW	Correlation register 3 – 15 adjustment factor (floating-point MSW and LSW)[22] as above

[22] The value of the input register corresponds to the values of the holding register at the same address 0x72xx.

			…	…	…	Reserved
			0x7220	r	float32 MSW	Lambda ratio adjustment factor (floating-point MSW)[21]
			0x7221	r	float32 LSW	Lambda ratio adjustment factor (floating-point LSW)[21]
			0x7222	r	float32 MSW	Pressure ratio adjustment factor (floating-point MSW)[21]
			0x7223	r	float32 LSW	Pressure ratio adjustment factor (floating-point LSW)[21]
			0x7224	r	float32 MSW	Flow ratio adjustment factor (floating-point MSW)[21]
			0x7225	r	float32 LSW	Flow ratio adjustment factor (floating-point LSW)[21]
			…	…	…	Reserved
			0x7300	r	float32 MSW	Lower measurement limit register 0 (floating-point MSW)
			0x7301	r	float32 LSW	Lower measurement limit register 0 (floating-point LSW)
			0x7302	r	float32 MSW	Lower measurement limit register 1 (floating-point MSW)
			0x7303	r	float32 LSW	Lower measurement limit register 1 (floating-point LSW)
			0x7304	r	float32 MSW	Lower measurement limit register 2 (floating-point MSW)
			0x7305	r	float32 LSW	Lower measurement limit register 2 (floating-point LSW)
			0x7306 – 0x731F	r	float32 MSW & float32 LSW	Lower measurement limit register 3 – 15 (floating-point MSW and LSW) as above
			…		…	Reserved
			0x7400	r	float32 MSW	Upper measurement limit register 0 (floating-point MSW)
			0x7401	r	float32 LSW	Upper measurement limit register 0 (floating-point LSW)
			0x7402	r	float32 MSW	Upper measurement limit register 1 (floating-point MSW)
			0x7403	r	float32 LSW	Upper measurement limit register 1 (floating-point LSW)
			0x7404	r	float32 MSW	Upper measurement limit register 2 (floating-point MSW)
			0x7405	r	float32 LSW	Upper measurement limit register 2 (floating-point LSW)
			0x7406 – 0x741F	r	float32 MSW & float32 LSW	Upper measurement limit register 3 – 15 (floating-point MSW and LSW) as above
			…		…	Reserved
		Administration				

			0x8000	r	UINT16	Actual chamber pressure in mbara
			0x8001	r	INT16	Actual chamber Temp. in $\frac{1}{10}$°C (if supported by pressure sensor, else 0xFFFF)
			0x8002	r	INT16	Actual CPU temperature in $\frac{1}{10}$°C
			0x8003	r	UINT16	Actual CPU supply voltage Vdda in mV
			0x8004	r	UINT16	Actual supply voltage +5V in mV
			0x8005	r	UINT16	Actual supply voltage +12V in mV
			0x8006	r	UINT32 MSW	Power-up counter MSW
			0x8007	r	UINT32 LSW	Power-up counter LSW
			0x8008	r	UINT32 MSW	Measurement counter MSW
			0x8009	r	UINT32 LSW	Measurement counter LSW
			0x800A	r	UINT32 MSW	Analysis counter MSW
			0x800B	r	UINT32 LSW	Analysis counter LSW
			0x800C	r	UINT32 MSW	Titan error counter MSW
			0x800D	r	UINT32 LSW	Titan error counter LSW
			0x800E	r	UINT16	Hardware revision code
			0x800F	r	UINT16	Ex zone information
			0x8010	r	UINT64 MSW	Titan chip SN bit 48..63
			0x8011	r	…	Titan chip SN bit 32..47
			0x8012	r	…	Titan chip SN bit 16..31
			0x8013	r	UINT64 LSW	Titan chip SN bit 0..15
			0x8014	r	UINT96 MSW	Microcontroller chip UID bit 80..95
			0x8015	r	…	Microcontroller chip UID bit 64..79
			0x8016	r	…	Microcontroller chip UID bit 48..63
			0x8017	r	…	Microcontroller chip UID bit 32..47
			0x8018	r	…	Microcontroller chip UID bit 16..31
			0x8019	r	UINT96 LSW	Microcontroller chip UID bit 0..15
			0x801A	r	UINT32 MSW	Device serial number bit 16..31
			0x801B	r	UINT32 LSW	Device serial number bit 0..15

			0x801C	r	UINT32 MSW	Operation-hour counter bit 16..31
			0x801D	r	UINT32 LSW	Operation-hour counter bit 0..15
			0x801E	r	UINT16	Number of available correlations
			…	…	…	Reserved
			0x8020	r	UINT32 MSW	Bootloader binary CRC[23] bit 16..31
			0x8021	r	UINT32 LSW	Bootloader binary CRC[23] bit 0..15
			0x8022	r	UINT32 MSW	Application binary CRC bit 16..31
			0x8023	r	UINT32 LSW	Application binary CRC bit 0..15
			0x8024	r	UINT32 MSW	Configuration binary CRC bit 16..31
			0x8025	r	UINT32 LSW	Configuration binary CRC bit 0..15
			0x8026	r	UINT32 MSW	Bootloader file header CRC[23] bit 16..31
			0x8027	r	UINT32 LSW	Bootloader file header CRC[23] bit 0..15
			0x8028	r	UINT32 MSW	Application file header CRC bit 16..31
			0x8029	r	UINT32 LSW	Application file header CRC bit 0..15
			0x802A	r	UINT32 MSW	Configuration file header CRC bit 16..31
			0x802B	r	UINT32 LSW	Configuration file header CRC bit 0..15
			…	…	…	Reserved
			0x8030	r	UINT16	Firmware version number: Major
			0x8031	r	UINT16	Firmware version number: Minor
			0x8032	r	UINT16	Firmware version number: Patch
			0x8033	r	UINT16	Reserved (returns always 0x0000)
			…	…	…	Reserved
			0x8200	r	UINT32 MSW	Adjustment mode MSW
			0x8201	r	UINT32 LSW	Adjustment mode LSW
			0x8202	r	UINT32 MSW	Adjustment reference MSW
			0x8203	r	UINT32 LSW	Adjustment reference LSW
			0x8204	r	UINT32 MSW	Adjustment cycles MSW

[23] Reports 0xFFFF for gasQS flonic NG devices.

			0x8205	r	UINT32 LSW	Adjustment cycles LSW
			0x8206	r	UINT32 MSW	Adjustment averages MSW
			0x8207	r	UINT32 LSW	Adjustment averages LSW
			0x8208	r	UINT32 MSW	Adjustment measurement cycles left MSW
			0x8209	r	UINT32 LSW	Adjustment measurement cycles left LSW
			0x820A	r	float32 MSW	Lambda ratio reference value (floating-point MSW)
			0x820B	r	float32 LSW	Lambda ratio reference value (floating-point LSW)
			0x820C	r	float32 MSW	Pressure ratio reference value (floating-point MSW)
			0x820D	r	float32 LSW	Pressure ratio reference value (floating-point LSW)
			0x820E	r	float32 MSW	Flow ratio reference value (floating-point MSW)
			0x820F	r	float32 LSW	Flow ratio reference value (floating-point LSW)
			…	…	…	Reserved
			0x8400	r	UINT16	Logfile data bytes at logfile pointer address offset 0 & 1
			0x8401	r	UINT16	Logfile data bytes at logfile pointer address offset 2 & 3
			…	…	…	…
			0x847B	r	UINT16	Logfile data bytes at logfile pointer address offset 246 & 247
			0x847C	r	UINT16	Logfile data bytes at logfile pointer address offset 248 & 249
			…	…	…	Reserved
			0x847E	r	UINT32 MSW	Logfile size (unsigned integer MSW)
			0x847F	r	UINT32 LSW	Logfile size (unsigned integer LSW)
			…	…	…	Reserved

Table 12: Input registers address mapping

6.8.5 Diagnostics (Function Code 0x2B)

Type	Function codes	Sub code	Address	Access type	Description
Diagnostics	0x08	0x00		w	Return query data
	0x08	0x03		w	Change ASCII input delimiter
	0x2B	0x0E		r	Read device information

Table 13: Diagnostics

6.8.6 Device Identification Returned by Function 0x2B, Sub-code 0x0E

Object ID	Description	gasQS example	Type	Category
0x00	VendorName	Same value as in 0x80 if not defined by customer	ASCII String	Basic
0x01	ProductCode	Same value as in 0x81 if not defined by customer	ASCII String	
0x02	MajorMinorRevision	Same value as in 0x82 if not defined by customer	ASCII String	
0x03	VendorURL	Same value as in 0x83 if not defined by customer	ASCII String	Regular
0x04	ProductName	Same value as in 0x84 if not defined by customer	ASCII String	
0x05	ModelName	Same value as in 0x85 if not defined by customer	ASCII String	
0x06	UserApplicationName	Same value as in 0x86 if not defined by customer	ASCII String	
0x07 … 0x7F	Reserved			
0x80	ManufacturerName	Mems AG	ASCII String	Extended / Private objects
0x81	OrderCode	Order code (Variable defined by config file)		
0x82	MajorMinorRevision	(Hardware revision x and Ex zone y)		
0x83	ManufacturerURL	www.mems.ch		
0x84	ProductName	gasQS(TM)		
0x85	ModelName	flonic		
0x86	UserApplicationName	xx.yy.zz (Firmware revision)		
0x87	PCB serial number	(Variable defined by config file)		
0x88	Device serial number	(Variable defined in EEPROM)		
0x89 … 0xBF	Reserved			
0xC0 … 0xCF	Correlation descriptions	(String defined by config file)		
0xD0 …	Reserved			

0xFF				

Table 14: Device Identification

6.8.6.1 Version numbering structure xx.yy.zz

xx major number or compatibility number

01, 02, 03, etc. up to 99 is incremented if compatibility with previous hardware is no longer given, or in the case of a new firmware development if, for example, major functional changes to a previous version are no longer feasible due to new increased requirements.

yy minor number - measurement principle index

Incremented when a change is made that has influence on the measurement principle.

zz patch number - functionality (& bugfix) index

The official name of this index is "functionality index". It is incremented when additional functionality is added or functionality (without metrological relevance) is changed, or a bug or problem is fixed.

6.8.6.2 Correlation description

Device identification object IDs from 0xC0 to 0xCF are related to the correlation registers 0 to 15 and return a description of the register's content. Descriptions are ASCII strings formatted in a machine-readable JSON object, containing a label for the register, the unit of the measurement value and the last measured numerical value. An example of the correlation description for a gasQS flonic NG device is shown in Table 15. The representation may differ depending on the device type and register definition (see Table 24).

Object ID	Correlation description example
0xC0	{"Label":"Counter","Unit":"","Value":4711}
0xC1	{"Label":"Pout","Unit":"mbara","Value":1024.3}
0xC2	{"Label":"Dn","Unit":"kg/m3","Value":1.234}
0xC3	{"Label":"Hs","Unit":"MJ/m3","Value":39.872}
0xC4	{"Label":"n/a","Unit":"","Value":null}
0xC5	{"Label":"n/a","Unit":"","Value":null}
0xC6	{"Label":"n/a","Unit":"","Value":null}
0xC7	{"Label":"Pm","Unit":"mbara","Value":30000}
0xC8	{"Label":"Tm","Unit":"K","Value":288.15
0xC9	{"Label":"Zn","Unit":"--","Value":0.997641}
0xCA	{"Label":"Z","Unit":"--","Value":0.943284}
0xCB	{"Label":"n/a","Unit":"","Value":null}
0xCC	{"Label":"n/a","Unit":"","Value":null}
0xCD	{"Label":"n/a","Unit":"","Value":null}
0xCE	{"Label":"n/a","Unit":"","Value":null}
0xCF	{"Label":"H2","Unit":"%","Value":0.000000}[24]

Table 15: Device Identification example for a standard gasQS flonic NG device.

[24] This correlation is non-metrology-controlled and optional. If not ordered it displays null.

Object ID	**Correlation description example**
0xC0	{"Label":"Counter","Unit":"","Value":4711}
0xC1	{"Label":"Pout","Unit":"mbara","Value":1024.3}
0xC2	{"Label":"Dn","Unit":"kg/m3","Value":1.234}
0xC3	{"Label":"Hs","Unit":"MJ/m3","Value":39.872}
0xC4	{"Label":"Wobbe","Unit":"MJ/m3","Value":53.450}
0xC5	{"Label":"H2","Unit":"%","Value":0.000000}
0xC6	{"Label":"n/a","Unit":"","Value":null}
0xC7	{"Label":"n/a","Unit":"","Value":null}
0xC8	{"Label":"n/a","Unit":"","Value":null}
0xC9	{"Label":"n/a","Unit":"","Value":null}
0xCA	{"Label":"n/a","Unit":"","Value":null}
0xCB	{"Label":"n/a","Unit":"","Value":null}
0xCC	{"Label":"n/a","Unit":"","Value":null}
0xCD	{"Label":"n/a","Unit":"","Value":null}
0xCE	{"Label":"n/a","Unit":"","Value":null}
0xCF	{"Label":"n/a","Unit":"","Value":null}

Table 16: Device Identification example for a custom gasQS flonic Z1 device.

Reading object IDs requires multiple accesses via the Modbus. If new measurement values are updated by the device between two Modbus accesses, it is not guaranteed that measurement values in the descriptions belong to the same measurement data set, especially when the device is operated in automatic measurement mode. To make sure the retrieved data set is consistent, the measurement values must be read from the Modbus input registers all at once.

6.9 Exception Codes

If an error occurs while the slave is processing the telegram received from the master, it responds with the received function code and appends a one-byte exception code as the data field. Here is the table of supported exception codes:

Code	Name	Meaning
01	ILLEGAL FUNCTION	The function code received in the query is not an allowable action for the server. This may be because the function code is only applicable to newer devices and was not implemented in the unit selected. It could also indicate that the server is in the wrong state to process a request of this type, for example because it is unconfigured and is being asked to return register values.
02	ILLEGAL DATA ADDRESS	The data address received in the query is not an allowable address for the server. More specifically, the combination of reference number and transfer length is invalid. For a controller with 100 registers, the PDU addresses the first register as 0, and the last one as 99. If a request is submitted with a starting register address of 96 and a quantity of registers of 4, then this request will successfully operate (address-wise at least) on registers 96, 97, 98, 99. If a request is submitted with a starting register address of 96 and a quantity of registers of 5, then this request will fail with Exception Code 0x02 “Illegal Data Address” since it attempts to operate on registers 96, 97, 98, 99 and 100, and there is no register with address 100.
03	ILLEGAL DATA VALUE	A value contained in the query data field is not an allowable value for the server. This indicates a fault in the structure of the remainder of a complex request, such as that the implied length is incorrect. It specifically does NOT mean that a data item submitted for storage in a register has a value outside the expectation of the application program, since the MODBUS protocol is unaware of the significance of any particular value of any particular register.
04	SERVER DEVICE FAILURE	An unrecoverable error occurred while the server was attempting to perform the requested action.
05	ACKNOWLEDGE	Specialized use in conjunction with programming commands. The server has accepted the request and is processing it, but a long duration of time will be required to do so. This response is returned to prevent a timeout error from occurring in the client. The client can next issue a Poll Program Complete message to determine if processing is completed.
06	SERVER DEVICE BUSY	Specialized use in conjunction with programming commands. The server is engaged in processing a long–duration program command. The client should retransmit the message later when the server is free.
08	MEMORY PARITY ERROR	Specialized use in conjunction with function codes 20 and 21 and reference type 6, to indicate that the extended file area failed to pass a consistency check. The server attempted to read the record file but detected a parity error in the memory. The client can retry the request, but service may be required.

Table 17: MODBUS Exception Codes

6.10 Internal Errors

6.10.1 Error Code Register

Internal device errors that occur during operation are stored in the error code input register at address 0x1000. This register holds the first error code that was recognized and will not be overwritten by subsequent errors until it is cleared by setting coil address 0x1000 to TRUE or during a measurement if the automatic error code reset is activated in holding register at address 0x8003. If an error or warning is manually reset before the current measurement sequence that caused the error or warning is completed, then subsequent errors will be reported. Correlation warnings (codes above 0x8000) can be overwritten by errors (codes below 0x8000).

Code	Name	Meaning
No exception occurred		
0x0000	EXCEPTION_NONE	No error occurred; the device is working properly.
µC hardware exceptions		
0x0001	EXCEPTION_ADC_ERROR	In case of error due to overrun when using ADC with DMA transfer.
0x0002	EXCEPTION_ADC_RECALIB_ERROR	ADC automatic self-calibration failed.
0x0003	EXCEPTION_I2C_ERROR	I2C communication error with the sensor or the EEPROM occurred.
0x0004	EXCEPTION_CLOCK_SOURCE_ERROR	Quartz oscillator is not working. The time source is not reliable.
0x0005	EXCEPTION_FLASH_CRC_ERROR	CRC check in Flash memory failed
Peripheral hardware exceptions		
0x1000	EXCEPTION_EEPROM_ACCESS_ERROR	EEPROM communication error occurred.
0x1001	EXCEPTION_TITAN_SENSOR_ERROR	Flow sensor communication error occurred.
0x1002	EXCEPTION_PRESSURE_SENSOR_ERROR	Invalid pressure sensor data.
0x1003	EXCEPTION_POWER_SUPPLY_ERROR	Power supply voltage error
Application exceptions		
0x2000	EXCEPTION_OS_UNKNOWN_ERROR	Unspecified OS error.
0x2001	EXCEPTION_OS_SIGNAL_TIMEOUT	Unspecified OS signal timeout.
0x2002	EXCEPTION_CALIBRATION_ERROR	The device does not have calibration data.
0x2003	EXCEPTION_VALVE_ERROR	Invalid valve configuration setting.
0x2004	EXCEPTION_UPSTREAM_PRESSURE	Upstream pressure level too low for measurement.
0x2005	EXCEPTION_DOWNSTREAM_PRESSURE	Downstream pressure level too high.
0x2006	EXCEPTION_NOT_READY	Pressure range not ready for data evaluation.
0x2007	EXCEPTION_MEASUREMENT_ERROR	Measurement data acquisition error.

0x2008	EXCEPTION_EVALUATION_ERROR	Measurement data evaluation error.
0x2009	EXCEPTION_PRESSURE_MODE_ERROR	Invalid pressure operation mode setting.
0x200A	EXCEPTION_NOZZLE_TIMEOUT_ERROR	Pressure drops too slow.
0x200B	EXCEPTION_LAMBDA_TIMEOUT_ERROR	Measurement data evaluation error.
0x200C	EXCEPTION_CONFIGURATION_ERROR	Invalid configuration data.
0x200D	EXCEPTION_ADJUSTMENT_ERROR	Invalid adjustment parameters.
Communication exceptions		
0x3000	EXCEPTION_MODBUS_ERROR	Internal Modbus driver error occurred.
Correlation warnings[25]		
0x8001	EXCEPTION_CORRELATION_ID_UNKNOWN	The specified correlation does not exist.
0x8002	EXCEPTION_CORRELATION_TEMP_WARNING	A temperature parameter is out of range.
0x8003	EXCEPTION_CORRELATION_PRESSURE_WARNING	A pressure parameter is out of range.
0x8004	EXCEPTION_CORRELATION_OVERRANGE_WARNING	The correlation result is too high and out of the calibrated range.
0x8005	EXCEPTION_CORRELATION_UNDERRANGE_WARNING	The correlation result is too low and out of the calibrated range.
0x8006	EXCEPTION_CORRELATION_RATIO_WARNING	The flow and pressure ratio could not be determined.
0x8007	EXCEPTION_CORRELATION_UPPER_LIMIT_WARNING	The correlation result is too high and out of the specified measurement range.
0x8008	EXCEPTION_CORRELATION_LOWER_LIMIT_WARNING	The correlation result is too low and out of the specified measurement range.
0x8009	EXCEPTION_ADJUSTMENT_PENDING_WARNING	The adjustment interval has expired and an adjustment is pending
0x800A	EXCEPTION_uC_TEMPERATURE_WARNING	Microcontroller temperature out of specified range.
0x800B	EXCEPTION_TITAN_TEMPERATURE_WARNING	Flow sensor temperature out of specified range.
0x800C	EXCEPTION_INPUT_PARAMETER_WARNING	A required input parameter (e.g. P_m or T_m) is not set. The corresponding correlation result (e.g. Z_m) will result in a NaN.

Table 18: gasQS flonic device error codes

[25] Warnings: The device is working properly but the measurement input parameters or the results might be out of range.

6.10.2 Error code reference

The error code reference is a time stamp based on the settings in holding register address 0x8004. Since the device does not have a real time clock, the selectable sources are: measurement counter, analysis counter and the operational hours counter. The measurement and analysis counters will not increment in case of an error. They show the count of the last successful measurement before a problem occurred.

6.10.3 Error counters

While the error code register holds the first occurring exception, the error counters will register all errors, also subsequent errors. In case of an overrun the counters will stay at a maximum value of 65535 and will not restart from zero.

Besides the error counters for every single error code, there is also a total errors counter for the sum of all errors. Error counters can be reset to zero by coil address 0x8005.

6.11 Adjustment Procedure Alarms

6.11.1 Alarm Status Flags

The adjustment alarm status flags are located at the Input Register addresses 0x1200 and 0x1201.

Bit No.	Name	Meaning
Control flags		
BIT_0	ADJUSTMENT_PENDING	A new adjustment procedure is required.
BIT_1	ADJUSTMENT_RUNNING	An adjustment procedure is currently running.
BIT_2	ADJUSTMENT_FAILED	The last adjustment procedure has failed.[21]
Adjustment error flags		
BIT_3	ADJUSTMENT_MEAS_ERR	General error during measurement acquisition.
BIT_4	ADJUSTMENT_STABILITY	Average stability error
BIT_5	ADJUSTMENT_HALF_MPE	Deviation between two adjustments exceeded by a half MPE.
BIT_6	ADJUSTMENT_MAX_DRIFT	Maximum deviation between two adjustments exceeded.
BIT_7	ADJUSTMENT_MAX_OFFSET	Maximum total deviation exceeded.
BIT_8	ADJUSTMENT_UPPER_LIMIT	Upper measurement limit exceeded.
BIT_9	ADJUSTMENT_LOWER_LIMIT	Lower measurement limit exceeded.
Reserved flags		
…	…	…
Correlation register alarm flags		
BIT_16	ADJUSTMENT_REG_CORR_0	An alarm flag was raised for correlation register 0 during adjustment.
BIT_17	ADJUSTMENT_REG_CORR_1	An alarm flag was raised for correlation register 1 during adjustment.
…	…	…
BIT_31	ADJUSTMENT_REG_CORR_15	An alarm flag was raised for correlation register 15 during adjustment.

Table 19: Adjustment alarm flag register

The ADJUSTMENT_PENDING flag is set automatically after an adjustment interval has elapsed or via the corresponding coil. All other flags are reset at the start of the adjustment procedure and then set when the alarm occurs. After successful execution of the adjustment procedure, all flags are reset.

6.12 Emergency Recovery

In case of unknown or misconfigured serial MODBUS settings, access to the gasQS flonic device can be regained by repeatedly sending a startup key after switching the supply power on. For doing this, a terminal with default serial configuration (19200 baud, 8 data bits, parity EVEN, 1 stop bit) can be used. Set up a point-to-point connection to the device. After power on, there is a 1 second window, indicated by the lower LED, lighting up red, during which a startup key is accepted.

Startup key	Description
r	Boot into application and reset UART settings to their default values. The 1 second window is active when the lower LED lights up in red during boot-up.

Table 20: Power-on startup keys.

When the device is fully started up after such an emergency recovery, the only command replied via Modbus is the device identification function, other Modbus functions are ignored and replied with an error. After an additional power cycle, the device is fully operational again still with the default serial settings and all permitted Modbus functions are operable again.

6.13 Physical Layer

Data is transmitted over a 2-wires (twisted balanced pair) TIE/EIA-485 (RS485) serial line. All devices are connected in parallel to a trunk cable. In addition to the two transmission wires, a third wire is used for common. The optional 4-wires mode is not supported. To increase cable length, interference immunity and the number of nodes, repeaters can be used.

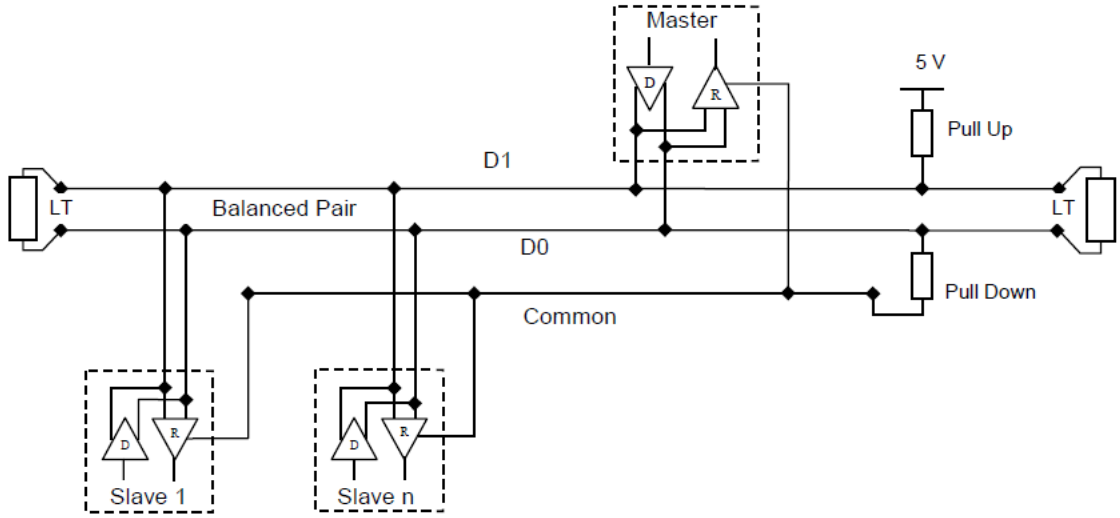

Figure 5: General 2-Wire Topology

The instrument can be used in both hazardous and non-hazardous environments. In the following chapters a distinction is made between the two application cases. Where there is no distinction, the chapter applies to both. For detailed information on switchgear in explosive atmospheres, refer to the relevant standards, including IEC 60079-0, IEC 60079-11 and IEC 60079-14.

6.13.1 Data Signalling Rates

9.6 and 19.2kbps are required, the latter being the default. 38.4kbps, 57.6kbps, 115.2kbps, 230.4kbps and 460.8kbps are supported. Transmission speed error must be less than 1%. The receiver must accept speed errors up to 2%.

6.13.2 Maximum number of devices without repeater

6.13.2.1 Hazardous areas

The physical interface is based on the specifications of the Profibus RS 485-IS User and Installation Guideline [3]. For intrinsic safe applications the guideline allows up to 32 devices. Please refer to the mentioned document for more details.

6.13.2.2 Non-hazardous areas

In non-hazardous applications the number of nodes can be tripled.

6.13.3 Topology

The instrument is designed to work in a linear structured bus, connected over a passive TAP to the trunk cable. The interface between the device and the Passive Tap, called IDv (Derivation Interface), should be as short as possible. Mems AG recommends connecting the Tap directly to the IDv-socket of the device, without using a derivation cable.

6.13.4 Data Rate and Bus Length

Theoretically the bus length can be up to 1 km as long as the data rate is less or equal to 100 kbit/s. If the data rate is further increased, the cable length must be reduced. For a data rate of 460.8 kbit/s the maximal length is 300 m.

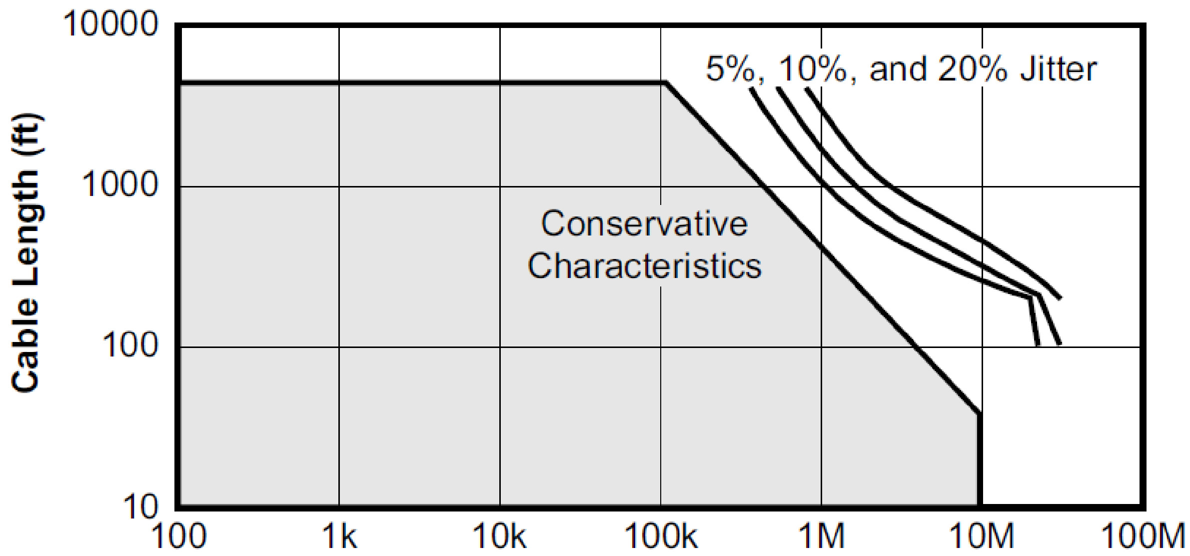

Figure 6: Data Rate vs. Bus Length [4]

The instrument is tested for cable lengths up to 30m. For bus length longer than 30m the compliance with EMC regulations is not guaranteed.

6.13.5 Stub Length

When connecting a node to the bus, the distance between the transceiver inputs and the cable trunk, known as the stub, should be as short as possible. As a general guideline, the electrical length, or round-trip delay, of a stub should be less than one-tenth of the rise time of the driver, thus giving a maximum physical stub length as shown in the equation below:

$$L_{stub} \leq 0.1 \cdot t_r \cdot v \cdot c = 0.1 \cdot 72ns \cdot v \cdot 3 \cdot 10^8 m/s$$

Where:

$t_r = estimated\ rise\ time\ of\ RS485\ driver\ in\ the\ gasQS\ flonic$

$c = speed\ of\ light$

$v = signal\ velocity\ of\ the\ cable\ as\ a\ factor\ of\ c$

6.13.6 Grounding Arrangements

The signal common must be connected directly to protective ground, preferably at one point only for the entire bus. Generally, this point chosen on the master device or on its tap.

6.13.7 Line Termination

To minimize the reflections from the end of the RS485-cable it is required to place a line termination near each of the 2 ends of the Bus. There is no line termination integrated into the device.
For standard termination, the termination resistor value is matched with the differential-mode characteristic impedance of the cabling. For twisted pair cables the impedance ranges from 100Ω to 150Ω. The Modbus Organization recommends a resistor value of 150Ω (0.5W).

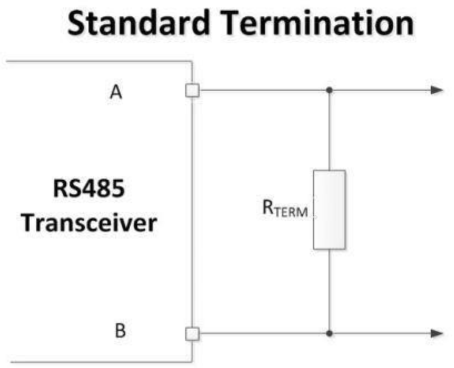

Figure 7: RS 485 standard termination [5]

If there are ground potential differences between nodes, high frequency noise can disturb the bus. To increase robustness of the system a split termination can be used. This technique uses two resistors that are equal to one half the characteristic impedance of the cable, with a capacitor placed to ground between the two resistors. This creates two low-pass filters for common mode noise on the RS485 bus line. The capacitor value is typically 4.7nF.

Using AC terminations helps alleviate this power dissipation without having as long of a bit-time requirement with respect to bus length. The Modbus Organization recommends a capacitor value of 1nF (≥10V) and a resistor value of 120Ω (0.25W). Warning, the RC network slows the rise and fall time of the differential signal and limits the maximum data rate of the network.

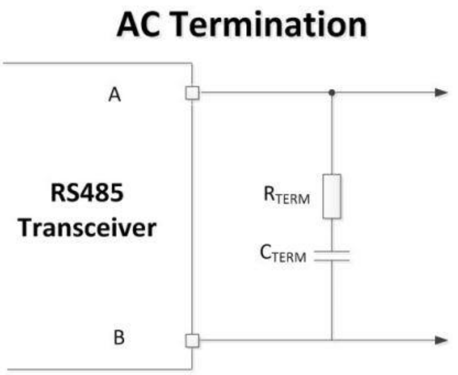

Figure 8: RS485 AC termination [5]

6.13.8 Line Polarization

The gasQS flonic does not provide a line polarization. The differential receiver used is “failsafe” to invalid bus states caused by:

- Open bus conditions, such as a disconnected connector
- Shorted bus conditions, such as cable damage shorting the twisted-pair together
- Idle bus conditions that occur when no driver on the bus is actively driving

The noise immunity of the receiver inputs during a bus fault condition is 130mVpp.

If one or several devices need polarization the pair must be implemented at one location for the whole serial bus. Generally, this point is chosen on the master device or on its Tap. Other devices must not implement any polarization. The maximum number of devices authorized on such a MODBUS serial line is reduced by 4 from a MODBUS without polarization.

6.13.9 Mechanical Interface

The gasQS flonic device uses a M12 A-coded female connector for the MODBUS interface.

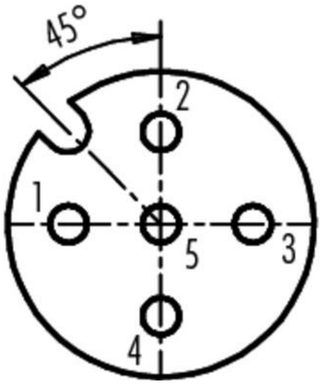

Figure 9: COM connector on gasQS device

1) SHLD
2) Not Connected
3) SGND
4) RS485-Data-
5) RS485-Data+

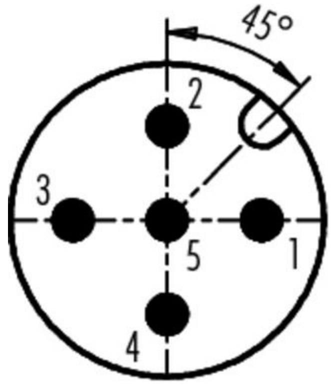

Figure 10: 5-pin male connector on cable

6.13.10 Cables

A MODBUS serial line cable must be shielded and use a balanced pair with a third wire for the Common. A Wire Gauge of AWG 24 (d = 0.51mm) is always sufficient for the MODBUS Data. Ensure that the characteristic impedance of the cable is higher than 100Ω.

The shell of the M12-A connector is directly connected to the housing and the shield wire of the cable.

6.13.10.1 Additional limiting safety values hazardous areas

Wire diameter:	> 0.1mm single wire for a fine-stranded conductor[26]
L/R ratio:	≤ 15µH/Ω for the lowest ambient temperature

To increase interference immunity, Mems AG recommends connecting the shield of the bus cable to earth at both ends. Attention for applications in hazardous areas, it must be ensured that potential equalization exists between the various earthing points (i.e. between the hazardous area and safe area). This must be ensured by installation and maintenance.

[26] In accordance with installation rules in EN 60079-14. The wire ends of fine-stranded conductors must be protected against separation of the strands, e.g. by means of cable lugs or core end sleeves.

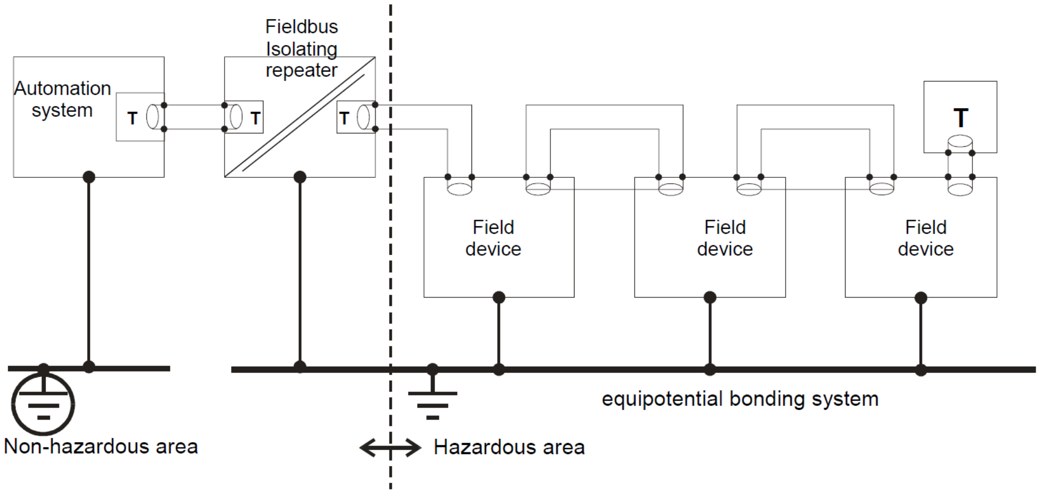

Figure 11: Ideal combination of shielding and earthing [3]

Please refer to chapter 3.5 of the Profibus RS 485-IS User and Installation Guideline [3] for more information about shielding and earthing.

6.13.10.2 Non-hazardous areas

At one end of each cable its shield must be connected to protective ground. If a connector is used at this end, the shell of the connector is connected to the shield of the cable.

To minimize errors in cabling, a colour code is recommended for the wires inside the Cable:

Signal Names	Recommended Colour
D1-TXD1	Orange
D0-TXD0	Brown
Common	Grey

Table 21: Recommended cable colour

6.13.11 Visual Diagnosis

For a visual diagnosis, communication status and device status must be indicated by LEDs:

LED	Colour	Value	State
Upper LED	Orange	Measurement	Switched ON: Device busy
		Adjustment	Flashing during adjustment procedure
	Red	Error	Flashing at 2Hz: Measurement failed or boundaries out of range
	Green	Device status	Switched ON: Device ready for measurement
Lower LED	Orange	Communication	Flashing during frame reception or sending
	Red	Error	Flashing at 2Hz: internal fault (communication fault or configuration error)
	Green	Device status	Switched ON: Device powered

Table 22: Visual diagnosis

7 Appendix

7.1 Password and Username Format

Reading and writing of passwords or usernames is done as ASCII strings via standard Modbus functions. Here two characters are written or read per holding register, and the end of the string is terminated with 0x00, if the number of characters is odd. Please note that big-endian format is used. 15 minutes after unlocking, the device is automatically locked again.

String	ASCII codes	Register values
"123"	0x31, 0x32, 0x33, 0x00	0x3132, 0x3300
"1234"	0x31, 0x32, 0x33, 0x34	0x3132, 0x3334

Table 23: ASCII string interpretation.

7.2 gasQS flonic Product Family Correlation Register Definitions

Register Adresses	flonic H-Gas	flonic NG-X	flonic NG-H (Option H_2)	flonic NG-L (Option H_2)	flonic Z1	*(preferred order)*
0x7000 & 0x7001	Analysis Counter	Analysis Counter	Analysis Counter	Analysis Counter	Analysis Counter	*Analysis Counter*
0x7002 & 0x7003	Pressure downstream	Pressure downstream	Pressure downstream	Pressure downstream	Sequence based on order code	*Pressure downstream*
0x7004 & 0x705	Density	Density	Density	Density		*Density*
0x7006 & 0x707	Higher Calorific Value	Higher Calorific Value	Higher Calorific Value	Higher Calorific Value		*Higher Calorific Value*
0x7008 & 0x709	Lower Calorific Value					*Lower Calorific Value*
0x700A & 0x700B	Higher Wobbe Index					*Higher Wobbe Index*
0x700C & 0x700D	Lower Wobbe Index					*Lower Wobbe Index*
0x700E & 0x700F		P_m	P_m	P_m		*P_m*
0x7010 & 0x7011		T_m	T_m	T_m		*T_m*
0x7012 & 0x7013		Z_n	Z_n	Z_n		*Z_n*
0x7014 & 0x7015		Z	Z	Z		*Z*
0x7016 & 0x7017						*K Factor*
0x7018 & 0x7019						*Conversion Factor (Z-Zahl)*
0x701A & 0x701B	Specific Gravity (AIR)					*Specific Gravity (AIR)*
0x701C & 0x701D	Methane Number (AVL)					*Methane Number (AVL)*
0x701E & 0x701F			H_2 content (0 - 23 mol%)	H_2 content (0 – 5 mol%)		*H_2 content (0 - 100 mol%)*

Table 24: gasQS flonic product family correlation register definitions.

7.3 Adjustment Procedure[27]

7.3.1 Adjustment Modes

Value	Name	Meaning
0	ADJUSTMENT_MODE_DISABLED	Adjustment not allowed.
1	ADJUSTMENT_MODE_MANUAL	Only manual adjustment allowed.
2	ADJUSTMENT_MODE_AUTO	Only automatic adjustment allowed.

Table 25: Adjustment mode input register values

7.3.1.1 Adjustment not allowed

Adjustment factors cannot be edited by the user. All adjustment factors are set to 1.0.

7.3.1.2 Only manual adjustment allowed

Adjustment factors can be edited by the user within a limited range of 0.9 to 1.1.

7.3.1.3 Only automatic adjustment allowed

Adjustment factors can be tuned by the automated adjustment procedure when the specified reference gas is used. Adjustment factors cannot be edited by the user.
If the adjustment procedure reports an excessive drift, the gasQS device won't display any further measurement values. The corresponding error code and the adjustment alarm flags are set and values are clearly recognizable as invalid (e.g., NaN will be returned).

7.3.2 Start adjustment procedure

To initiate an adjustment procedure, the device must be unlocked with the admin password. In addition, a username and a timestamp must be written to the device, otherwise the procedure cannot be started.

[27] The adjustment procedure is not available by default. This option needs to be preconfigured at the factory.

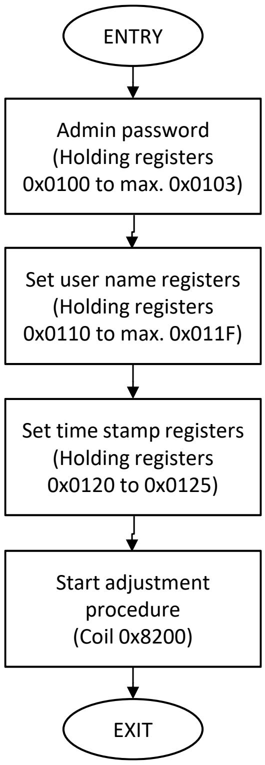

Figure 12: Start adjustment procedure.

7.3.3 Read adjustment logfile

The adjustment log is read out via a storage address (file pointer) and a range of input registers mapped to it. These register data can be interpreted as a byte stream. The beginning of the file always starts with the file pointer 0. The range is limited to the maximum quantity of 125 registers that can be read with the input register read function. The readout can be terminated when the file size (in bytes) is reached or when one or more EOF (0xFF) are read out. The file size can be determined via the corresponding input register. To read the logfile, administration rights are required, therefore requiring the device to be unlocked with the admin password.

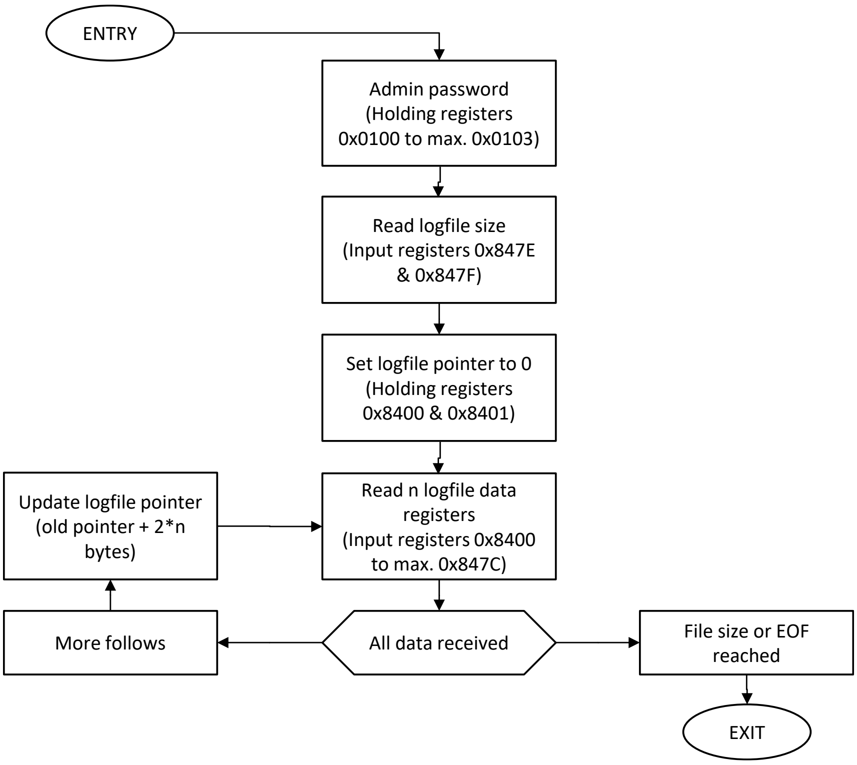

Figure 13: Read adjustment logfile.

7.4 Invalid Measurement Data Values

7.4.1 Not a Number (IEEE 754, NaN)

s111 1111 1qpp pppp pppp pppp pppp pppp

s: Signum (ignored)
q: Quiet flag (1 = quiet, 0 = signaling)
p: Payload

Special values
0111 1111 1000 0000 0000 0000 0000 0000 -> 0x7F800000 -> +Infinity
1111 1111 1000 0000 0000 0000 0000 0000 -> 0xFF800000 -> -Infinity

7.4.2 gasQS flonic 32-bit Floating Point NaN

s111 1111 1qnn nnnn pppp pppp pppp pppp = 0111 1111 1100 0000 pppp pppp pppp pppp

s: Signum (always 0 for the gasQS flonic device)
q: Quiet flag (always 1 = quiet for the gasQS flonic device)
n: Payload (ignored, always 0 for the gasQS flonic device)

p: Payload (16-bit Error code register value)

Special value interpretation for the gasQS flonic V2:
0111 1111 1100 0000 0000 0000 0000 0000 -> 0x7FC00000 -> Value not available (not configured).

7.4.3 16-bit Power of 10 Representation (Proprietary)

Normal interpretation of the register values: $y = a * 10^b$

Valid range for:
Factor a: -32768 to 32767
Exponent b: -63 to 32639

NaN interpretation for:
Exponent b: s111 1111 1q00 0000 = 0111 1111 1100 0000
Factor a: pppp pppp pppp pppp

s: Signum (always 0 for the gasQS flonic device)
q: Quiet flag (always 1 = quiet for the gasQS flonic device)
p: Payload (16-bit Error code register value)

8 Abbreviations and Definitions

ADU .... Application Data Unit
ASCII .... American Standard Code for Information Interchange
CAN .... Controller Area Network
JSON .... JavaScript Object Notation
LSB .... Least Significant Bit
LSW .... Least Significant Word (16-bit data)
MPE .... Maximum permissible error
MSB .... Most Significant Bit
MSW .... Most Significant Word (16-bit data)
NaN .... Not a Number
NOP .... No operation
PDU .... Protocol Data Unit
r .... Read access
rp .... Read access protected
RFC .... Request for Comments
RTU .... Remote Terminal Unit
UART .... Universal Asynchronous Receiver Transmitter
w .... Write access
wp .... Write access protected

9 References

[1] Modbus Organization, “Technical Resources - Modbus Serial Line Protocol and Implementation Guide,” 20 December 2006. [Online]. Available: http://modbus.org/docs/Modbus_over_serial_line_V1_02.pdf. [Accessed 2017 December 2017].

[2] Modbus Organization, “Technical Resources - MODBUS Application Protocol Specification,” 26 April 2012. [Online]. Available: http://www.modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf. [Accessed 22 December 2017].

[3] PROFIBUS Nutzerorganisation e.V., “Profibus RS 485-IS User and Installation Guideline,” PROFIBUS Nutzerorganisation e.V., Karlsruhe, 2003.

[4] Texas Instruments Incorporated, “Datasheet SN65HVD78,” December 2016. [Online]. Available: http://www.ti.com/lit/ds/symlink/sn65hvd78.pdf. [Accessed 4 September 2018].

[5] J. Griffith, “RS-485 basics: When termination is necessary, and how to do it properly,” Texas Instruments Incorporated, 28 July 2016. [Online]. Available: https://e2e.ti.com/blogs_/b/analogwire/archive/2016/07/28/rs-485-basics-when-termination-is-necessary-and-how-to-do-it-properly. [Accessed 23 November 2017].